

Pengenalan Defuzifikasi

Maksud dan Tujuan Defuzifikasi

Dalam sistem kendali logika fuzzy, proses inferensi fuzzy menghasilkan *output* berupa himpunan fuzzy. Himpunan fuzzy ini merepresentasikan solusi atau keputusan dalam bentuk yang tidak pasti atau 'kabur'. Namun, sistem kendali pada dunia nyata, seperti aktuator atau sistem pengambilan keputusan otomatis, memerlukan nilai *crisp* atau nilai tunggal yang jelas dan pasti untuk dapat beroperasi.

Defuzifikasi adalah proses konversi himpunan fuzzy menjadi nilai *crisp*. Proses ini menjembatani kesenjangan antara *output* fuzzy dari sistem inferensi dengan kebutuhan input berupa nilai tunggal pada sistem kendali atau aplikasi praktis. Dengan kata lain, defuzifikasi memastikan bahwa hasil dari logika fuzzy dapat diimplementasikan dan digunakan dalam sistem yang memerlukan keputusan yang konkret dan terukur.

Jenis Metode dan Dasar Teorinya

Terdapat berbagai metode defuzifikasi yang dapat digunakan, masing-masing dengan pendekatan dan karakteristiknya sendiri. Pemilihan metode defuzifikasi dapat mempengaruhi kinerja sistem kendali secara keseluruhan. Berikut adalah beberapa metode defuzifikasi yang umum digunakan:

1. Metode Centroid (Pusat Area)

Metode Centroid, juga dikenal sebagai metode pusat gravitasi, adalah metode defuzifikasi yang paling populer dan luas digunakan. Metode ini menghitung nilai *crisp* sebagai pusat area di bawah kurva fungsi keanggotaan himpunan fuzzy. Metode ini mempertimbangkan seluruh bentuk himpunan fuzzy, sehingga cenderung memberikan hasil yang stabil dan representatif.

Secara matematis, nilai defuzifikasi centroid (

$$z_{\text{centroid}}$$

) dihitung menggunakan rumus:

$$z_{\text{centroid}} = \frac{\int z \cdot \mu(z) dz}{\int \mu(z) dz}$$

dimana: -

$$z$$

adalah semesta pembicaraan (universe of discourse). -

$$\mu(z)$$

adalah fungsi keanggotaan himpunan fuzzy. -

$$\int$$

menunjukkan operasi integral terhadap semesta pembicaraan.

2. Metode Bisector (Pembagi Dua)

Metode Bisector mencari nilai pada semesta pembicaraan yang membagi area di bawah fungsi keanggotaan menjadi dua bagian yang sama luas. Dengan kata lain, metode ini mencari garis vertikal yang membagi area himpunan fuzzy menjadi dua area yang sama besar.

Nilai defuzifikasi bisector (

$$z_{\text{bisector}}$$

) diperoleh dengan menyelesaikan persamaan:

$$\int_{-\infty}^{z_{\text{bisector}}} \mu(z) dz = \int_{z_{\text{bisector}}}^{\infty} \mu(z) dz$$

3. Metode Mean of Maximum (MOM) (Rata-rata Maksimum)

Metode Mean of Maximum (MOM) menghitung nilai *crisp* sebagai rata-rata dari semua nilai pada semesta pembicaraan yang memiliki derajat keanggotaan maksimum. Metode ini berfokus pada wilayah dengan derajat keanggotaan tertinggi dan mengabaikan bentuk keseluruhan dari himpunan fuzzy.

Nilai defuzifikasi MOM (

$$z_{\text{mom}}$$

) dihitung menggunakan rumus:

$$z_{\text{mom}} = \frac{\sum_i z_i}{N}$$

dimana: -

$$z_i$$

adalah nilai-nilai pada semesta pembicaraan yang memiliki derajat keanggotaan maksimum. -

$$N$$

adalah jumlah total nilai

$$z_i$$

4. Metode Smallest of Maximum (SOM) (Nilai Maksimum Terkecil)

Metode Smallest of Maximum (SOM) memilih nilai *crisp* sebagai nilai terkecil di antara semua nilai pada semesta pembicaraan yang memiliki derajat keanggotaan maksimum. Metode ini cenderung menghasilkan nilai yang lebih konservatif dibandingkan dengan MOM atau LOM.

Nilai defuzifikasi SOM (

$$z_{\text{som}}$$

) dihitung menggunakan rumus:

$$z_{\text{som}} = \min(z_i)$$

dimana

$$z_i$$

adalah nilai-nilai pada semesta pembicaraan yang memiliki derajat keanggotaan maksimum.

5. Metode Largest of Maximum (LOM) (Nilai Maksimum Terbesar)

Metode Largest of Maximum (LOM) memilih nilai *crisp* sebagai nilai terbesar di antara semua nilai pada semesta pembicaraan yang memiliki derajat keanggotaan maksimum. Metode ini menghasilkan nilai yang lebih agresif dibandingkan dengan SOM atau MOM.

Nilai defuzifikasi LOM (

$$z_{\text{lom}}$$

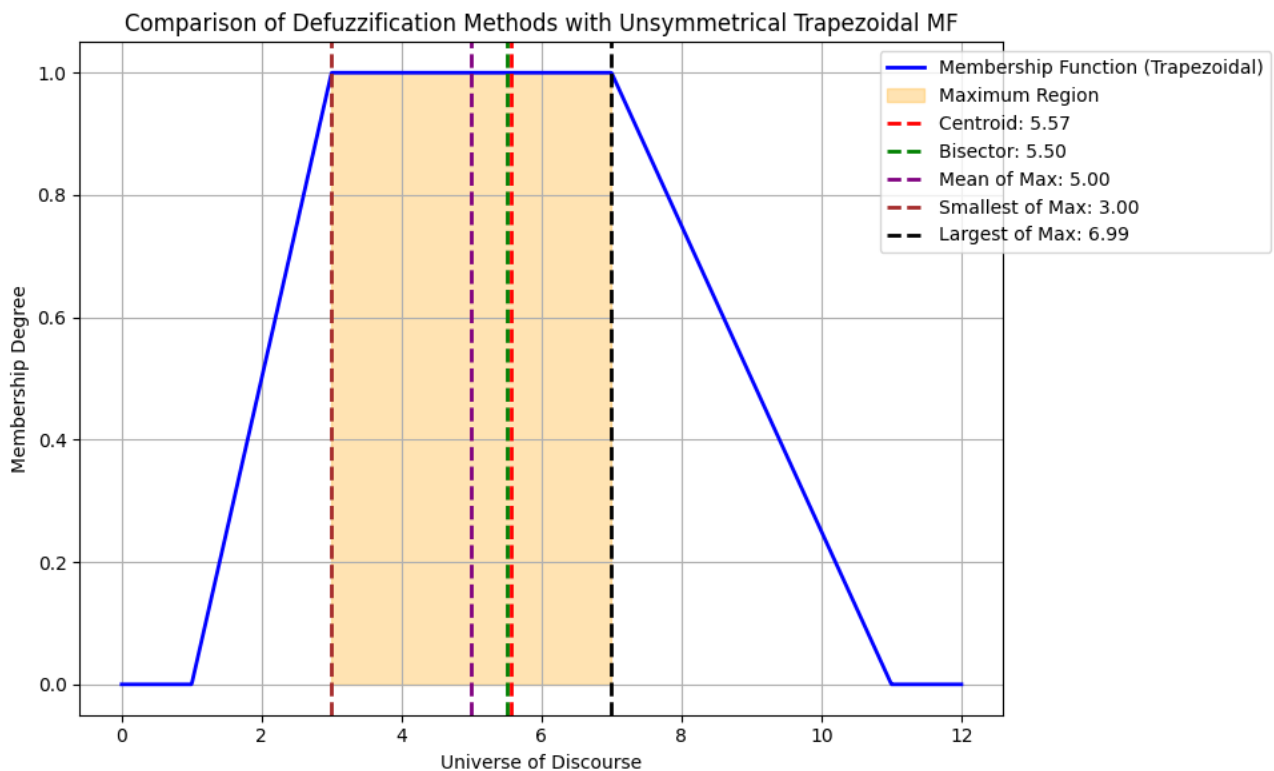
) dihitung menggunakan rumus:

$$z_{\text{lom}} = \max(z_i)$$

dimana

$$z_i$$

adalah nilai-nilai pada semesta pembicaraan yang memiliki derajat keanggotaan maksimum.



Penjelasan Detail Jenis Metode Defuzzifikasi

1. Metode Centroid (Pusat Area)

Metode Centroid adalah salah satu teknik defuzzifikasi yang paling umum digunakan dalam sistem logika fuzzy. Metode ini menghitung pusat gravitasi (atau centroid) dari area di bawah kurva fungsi keanggotaan. Metode ini sangat disukai karena memberikan nilai *crisp* yang seimbang dan representatif, berdasarkan pada keseluruhan himpunan fuzzy.

Konsep Dasar

Metode centroid menentukan nilai defuzzifikasi

$$z_{\text{centroid}}$$

sebagai rata-rata tertimbang dari semua nilai yang mungkin dalam semesta pembicaraan, di mana bobotnya ditentukan oleh derajat keanggotaan himpunan fuzzy. Secara matematis, ini dinyatakan sebagai:

$$z_{\text{centroid}} = \frac{\int x \cdot \mu(x) dx}{\int \mu(x) dx}$$

Dimana: -

$$x$$

: Semesta pembicaraan (rentang input). –

$$\mu(x)$$

: Fungsi keanggotaan yang mendefinisikan derajat keanggotaan untuk setiap

$$x$$

. –

$$\int x \cdot \mu(x) dx$$

: Pembilang merepresentasikan “momen” dari area di bawah kurva. –

$$\int \mu(x) dx$$

: Penyebut merepresentasikan total area di bawah kurva.

Hasil

$$z_{\text{centroid}}$$

adalah nilai *crisp* yang paling baik merepresentasikan himpunan fuzzy.

Aspek–aspek Penting yang Perlu Diperhatikan

1. **Rata-rata Tertimbang**: Metode centroid mempertimbangkan keseluruhan fungsi keanggotaan, membuatnya sensitif terhadap bentuk dan distribusi himpunan fuzzy.
2. **Robustness (Ketahanan)**: Metode ini bekerja dengan baik untuk fungsi keanggotaan simetris maupun asimetris.
3. **Interpretasi**: Nilai centroid dapat dianggap sebagai “pusat massa” dari himpunan fuzzy.
4. **Akurasi**: Memberikan representasi yang presisi dari himpunan fuzzy.
5. **Fleksibilitas**: Dapat digunakan untuk berbagai bentuk fungsi keanggotaan (segitiga, trapezium, Gaussian, dll.).
6. **Interpretasi Fisik**: Merepresentasikan “pusat gravitasi,” sehingga mudah dipahami secara intuitif.
7. **Intensitas Komputasi**: Memerlukan integrasi, yang bisa menjadi mahal secara komputasi untuk fungsi keanggotaan yang kompleks.
8. **Sensitivitas terhadap Outlier**: Jika fungsi keanggotaan memiliki ekor yang panjang, centroid dapat bergeser menjauhi wilayah utama yang diminati.

Langkah–langkah Menghitung Centroid

1. Definisikan Fungsi Keanggotaan:

- Tentukan semesta pembicaraan (

$$x$$

) dan fungsi keanggotaan (

$$\mu(x)$$

).

2. Hitung Pembilang:

- Hitung

$$\int x \cdot \mu(x) dx$$

, yaitu jumlah tertimbang dari semua nilai

$$x$$

berdasarkan derajat keanggotaannya.

3. Hitung Penyebut:

- Hitung

$$\int \mu(x) dx$$

, yaitu total area di bawah fungsi keanggotaan.

4. Bagi Hasil Pembilang dengan Penyebut:

- Bagi nilai pembilang dengan penyebut untuk mendapatkan nilai centroid.

Kelebihan

1. **Akurasi:** Memberikan representasi yang presisi dari himpunan fuzzy.
2. **Versatilitas:** Bekerja untuk berbagai bentuk fungsi keanggotaan (segitiga, trapesium, Gaussian, dll.).
3. **Interpretasi Fisik:** Merepresentasikan “pusat gravitasi,” membuatnya intuitif.

Kekurangan

1. **Intensitas Komputasi:** Memerlukan integrasi, yang bisa menjadi mahal secara komputasi untuk fungsi keanggotaan yang kompleks.
2. **Sensitivitas terhadap Outlier:** Jika fungsi keanggotaan memiliki ekor yang panjang, centroid dapat bergeser menjauhi wilayah utama yang diminati.

Aplikasi

Metode centroid umum digunakan dalam: – Sistem kendali fuzzy (misalnya, pengontrol suhu, regulator kecepatan). – Sistem pengambilan keputusan. – Tugas pengenalan pola dan klasifikasi.

Dengan memahami metode centroid, mahasiswa dapat mengapresiasi perannya dalam mengkonversi *output* fuzzy menjadi nilai *crisp* yang dapat ditindaklanjuti, sambil mempertahankan integritas proses penalaran fuzzy.

Contoh Perhitungan Matematika Defuzifikasi Centroid

1. Definisi Fungsi Keanggotaan

Fungsi keanggotaan trapesium asimetris didefinisikan sebagai:

$$\mu_{\text{trap}}(x) = \begin{cases} 0 & \text{jika } x \leq 1, \\ \frac{x-1}{3-1} = \frac{x-1}{2} & \text{jika } 1 < x \leq 3, \\ 1 & \text{jika } 3 < x \leq 7, \\ \frac{11-x}{11-7} = \frac{11-x}{4} & \text{jika } 7 < x \leq 11, \\ 0 & \text{jika } x > 11. \end{cases}$$

2. Rumus Centroid

Nilai centroid

$$z_{\text{centroid}}$$

dihitung sebagai:

$$z_{\text{centroid}} = \frac{\int x \cdot \mu_{\text{trap}}(x) dx}{\int \mu_{\text{trap}}(x) dx}.$$

Kita akan menghitung pembilang (

$$\int x \cdot \mu_{\text{trap}}(x) dx$$

) dan penyebut (

$$\int \mu_{\text{trap}}(x) dx$$

) secara terpisah.

3. Pecah menjadi Wilayah

Fungsi keanggotaan trapesium dapat dibagi menjadi empat wilayah:

1. **Wilayah 1:**

$$1 < x \leq 3$$

(kenaikan linear).

2. **Wilayah 2:**

$$3 < x \leq 7$$

(puncak datar).

3. **Wilayah 3:**

$$7 < x \leq 11$$

(penurunan linear).

4. **Wilayah 4:** Di luar rentang ini,

$$\mu_{\text{trap}}(x) = 0$$

4. Hitung Penyebut (

$$\int \mu_{\text{trap}}(x) dx$$

)**Wilayah 1:**

$$1 < x \leq 3$$

$$\begin{aligned} \int_1^3 \mu_{\text{trap}}(x) dx &= \int_1^3 \frac{x-1}{2} dx = \frac{1}{2} \int_1^3 (x-1) dx \\ &= \frac{1}{2} \left[\frac{(x-1)^2}{2} \right]_1^3 = \frac{1}{2} \left[\frac{(3-1)^2}{2} - \frac{(1-1)^2}{2} \right] = \frac{1}{2} \cdot \frac{4}{2} = 1. \end{aligned}$$

Wilayah 2:

$$3 < x \leq 7$$

$$\int_3^7 \mu_{\text{trap}}(x) dx = \int_3^7 1 dx = [x]_3^7 = 7 - 3 = 4.$$

Wilayah 3:

$$7 < x \leq 11$$

$$\begin{aligned} \int_7^{11} \mu_{\text{trap}}(x) dx &= \int_7^{11} \frac{11-x}{4} dx = \frac{1}{4} \int_7^{11} (11-x) dx \\ &= \frac{1}{4} \left[11x - \frac{x^2}{2} \right]_7^{11} = \frac{1}{4} \left[\left(11(11) - \frac{11^2}{2} \right) - \left(11(7) - \frac{7^2}{2} \right) \right] \\ &= \frac{1}{4} \left[\left(121 - \frac{121}{2} \right) - \left(77 - \frac{49}{2} \right) \right] = \frac{1}{4} \left[\frac{121}{2} - \frac{154 - 49}{2} \right] = \frac{1}{4} \left[\frac{121}{2} - \frac{105}{2} \right] = \frac{1}{4} \cdot \frac{16}{2} = 2. \end{aligned}$$

Total Penyebut

$$\int \mu_{\text{trap}}(x) dx = 1 + 4 + 2 = 7.$$

5. Hitung Pembilang (

$$\int x \cdot \mu_{\text{trap}}(x) dx$$

)**Wilayah 1:**

$$1 < x \leq 3$$

$$\begin{aligned}
\int_1^3 x \cdot \mu_{\text{trap}}(x) dx &= \int_1^3 x \cdot \frac{x-1}{2} dx = \frac{1}{2} \int_1^3 x(x-1) dx \\
&= \frac{1}{2} \int_1^3 (x^2 - x) dx = \frac{1}{2} \left[\frac{x^3}{3} - \frac{x^2}{2} \right]_1^3 \\
&= \frac{1}{2} \left[\left(\frac{3^3}{3} - \frac{3^2}{2} \right) - \left(\frac{1^3}{3} - \frac{1^2}{2} \right) \right] = \frac{1}{2} \left[\left(9 - \frac{9}{2} \right) - \left(\frac{1}{3} - \frac{1}{2} \right) \right] \\
&= \frac{1}{2} \left[\frac{9}{2} - \left(\frac{2-3}{6} \right) \right] = \frac{1}{2} \left[\frac{9}{2} + \frac{1}{6} \right] = \frac{1}{2} \left[\frac{27+1}{6} \right] = \frac{1}{2} \cdot \frac{28}{6} = \frac{14}{6} = \frac{7}{3}.
\end{aligned}$$

Wilayah 2:

$$3 < x \leq 7$$

$$\begin{aligned}
\int_3^7 x \cdot \mu_{\text{trap}}(x) dx &= \int_3^7 x \cdot 1 dx = \int_3^7 x dx = \left[\frac{x^2}{2} \right]_3^7 \\
&= \frac{7^2}{2} - \frac{3^2}{2} = \frac{49}{2} - \frac{9}{2} = \frac{40}{2} = 20.
\end{aligned}$$

Wilayah 3:

$$7 < x \leq 11$$

$$\begin{aligned}
\int_7^{11} x \cdot \mu_{\text{trap}}(x) dx &= \int_7^{11} x \cdot \frac{11-x}{4} dx = \frac{1}{4} \int_7^{11} x(11-x) dx \\
&= \frac{1}{4} \int_7^{11} (11x - x^2) dx = \frac{1}{4} \left[\frac{11x^2}{2} - \frac{x^3}{3} \right]_7^{11} \\
&= \frac{1}{4} \left[\left(\frac{11(11^2)}{2} - \frac{11^3}{3} \right) - \left(\frac{11(7^2)}{2} - \frac{7^3}{3} \right) \right] \\
&= \frac{1}{4} \left[\left(\frac{1331}{2} - \frac{1331}{3} \right) - \left(\frac{539}{2} - \frac{343}{3} \right) \right] \\
&= \frac{1}{4} \left[\frac{3993 - 2662}{6} - \frac{1617 - 686}{6} \right] = \frac{1}{4} \left[\frac{1331}{6} - \frac{931}{6} \right] = \frac{1}{4} \cdot \frac{400}{6} = \frac{100}{6} = \frac{50}{3}.
\end{aligned}$$

Total Pembilang

$$\int x \cdot \mu_{\text{trap}}(x) dx = \frac{7}{3} + 20 + \frac{50}{3} = \frac{7+50}{3} + 20 = \frac{57}{3} + 20 = 19 + 20 = 39.$$

6. Hitung Nilai Centroid

Akhirnya, nilai centroid adalah:

$$z_{\text{centroid}} = \frac{\int x \cdot \mu_{\text{trap}}(x) dx}{\int \mu_{\text{trap}}(x) dx} = \frac{39}{7} \approx 5.57.$$

Hasil Akhir

Nilai centroid yang dihitung sesuai dengan keluaran kode program:

$$z_{\text{centroid}} \approx 5.57.$$

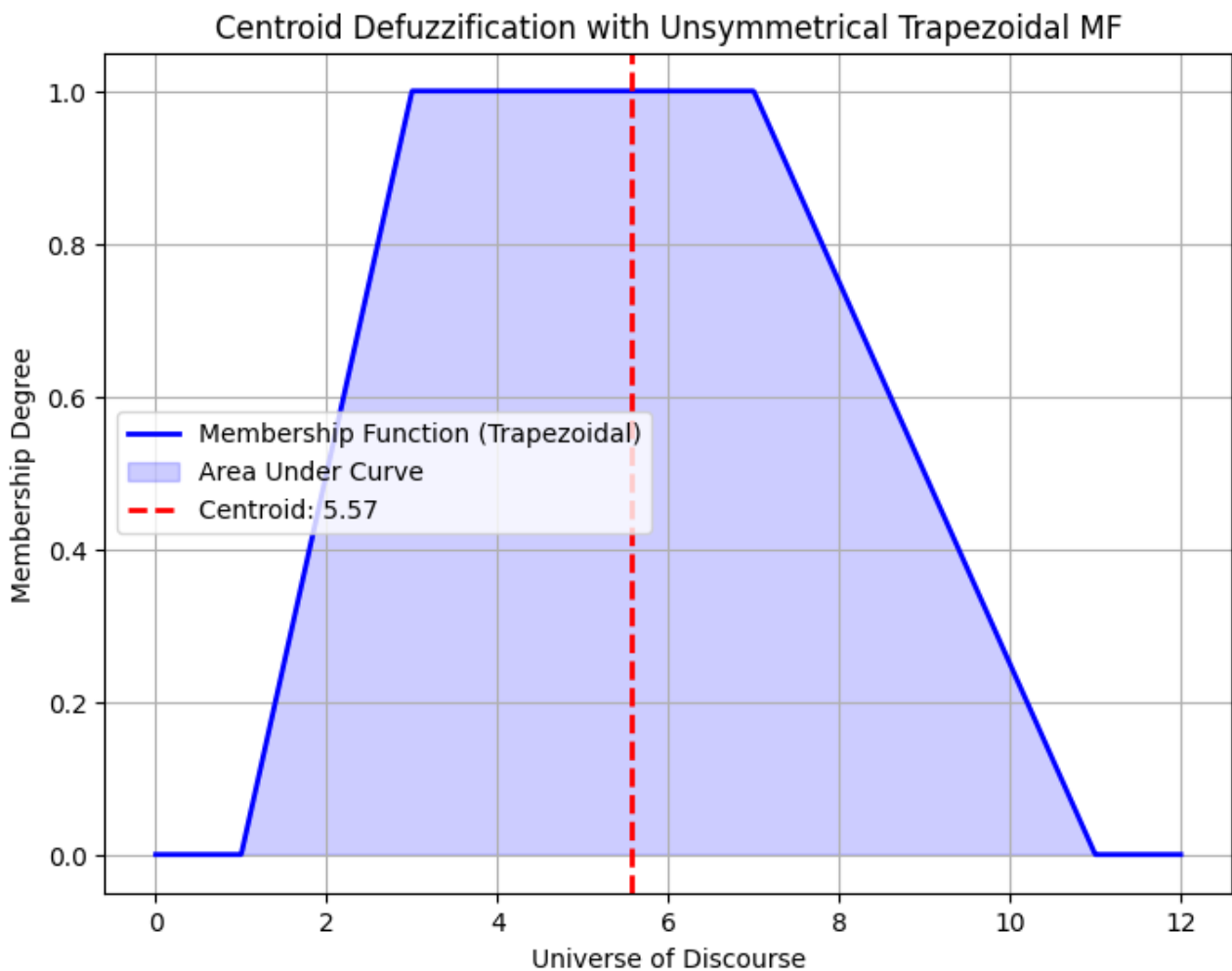
Contoh Kode Program

Berikut adalah contoh kode program Python yang mengimplementasikan metode centroid untuk defuzifikasi menggunakan fungsi keanggotaan trapesium asimetris.

```

1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
5
6 # Langkah 1: Definisikan semesta pembicaraan
7 x = np.linspace(0, 12, 1000)
8
9 # Langkah 2: Definisikan fungsi keanggotaan trapesium asimetris
10 mfx = fuzz.trapmf(x, [1, 3, 7, 11])
11
12 # Langkah 3: Lakukan defuzifikasi centroid
13 centroid_value = defuzz(x, mfx, mode='centroid')
14
15 # Langkah 4: Visualisasikan fungsi keanggotaan, nilai centroid, dan area di
    bawah kurva
16 plt.figure(figsize=(8, 6))
17
18 # Plot fungsi keanggotaan
19 plt.plot(x, mfx, label='Fungsi Keanggotaan (Trapezium)', color='biru',
    linewidth=2)
20
21 # Isi area di bawah kurva untuk menunjukkan area yang digunakan dalam
    perhitungan centroid
22 plt.fill_between(x, mfx, color='biru', alpha=0.2, label='Area Di Bawah Kurva')
23
24 # Tandai nilai centroid dengan garis vertikal putus-putus
25 plt.axvline(x=centroid_value, color='merah', linestyle='--', linewidth=2,
    label=f'Centroid: {centroid_value:.2f}')
26
27 # Tambahkan label, judul, legenda, dan grid
28 plt.title('Defuzifikasi Centroid dengan Fungsi Keanggotaan Trapezium Asimetris')
29 plt.xlabel('Semesta Pembicaraan')
30 plt.ylabel('Derajat Keanggotaan')
31 plt.legend()
32 plt.grid(True)
33
34 # Tampilkan plot
35 plt.show()
36
37 # Cetak nilai centroid (opsional)
38 print(f"Nilai Centroid: {centroid_value:.2f}")

```



```
1 Nilai Centroid: 5.57
```

Penjelasan Kode Program

```
1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
```

1. Import Library:

- **numpy**: Digunakan untuk komputasi numerik, seperti mendefinisikan semesta pembicaraan (x) dan melakukan operasi matematika.
- **skfuzzy**: Library untuk operasi logika fuzzy. Menyediakan alat untuk membuat fungsi keanggotaan dan melakukan defuzifikasi.
- **matplotlib.pyplot**: Digunakan untuk memvisualisasikan fungsi keanggotaan dan hasil defuzifikasi.

```
1 # Langkah 3: Lakukan defuzifikasi centroid
2 centroid_value = defuzz(x, mfx, mode='centroid')
```

2. Lakukan Defuzifikasi Centroid:

- Fungsi `defuzz` dari `skfuzzy.defuzzify` digunakan untuk menghitung nilai centroid.
- `mode='centroid'` menentukan bahwa metode centroid digunakan.
- Metode centroid menghitung pusat gravitasi dari area di bawah kurva fungsi keanggotaan:

$$z_{\text{centroid}} = \frac{\int x \cdot \mu(x) \, dx}{\int \mu(x) \, dx}.$$

- Baris kode ini adalah inti dari proses defuzifikasi centroid dalam program ini. Fungsi `defuzz` mengambil semesta pembicaraan (x), fungsi keanggotaan ($\mu(x)$), dan metode defuzifikasi ('centroid') sebagai input, dan menghasilkan nilai *crisp* centroid.

Tentu, berikut adalah draft bagian *handout* mengenai Metode Bisector dalam bahasa Indonesia, mengikuti format yang telah digunakan untuk Metode Centroid:

2. Metode Bisector (Pembagi Dua)

Metode Bisector adalah teknik defuzifikasi yang digunakan dalam sistem logika fuzzy untuk menghitung nilai *crisp* dari himpunan fuzzy. Metode ini menentukan garis vertikal yang membagi area di bawah fungsi keanggotaan menjadi dua bagian yang sama luas. Metode ini sangat berguna ketika fungsi keanggotaan bersifat asimetris atau tidak beraturan, karena menyeimbangkan distribusi himpunan fuzzy.

Konsep Dasar

Metode bisector mencari nilai

$$x$$

(

$$z_{\text{bisector}}$$

) sedemikian rupa sehingga area di bawah fungsi keanggotaan di sebelah kiri

$$z_{\text{bisector}}$$

sama dengan area di sebelah kanan

$$z_{\text{bisector}}$$

. Secara matematis, ini dinyatakan sebagai:

$$\int_{-\infty}^{z_{\text{bisector}}} \mu(x) \, dx = \int_{z_{\text{bisector}}}^{\infty} \mu(x) \, dx$$

Dimana: -

$$x$$

: Semesta pembicaraan (rentang input). –

$$\mu(x)$$

: Fungsi keanggotaan yang mendefinisikan derajat keanggotaan untuk setiap

$$x$$

. –

$$z_{\text{bisector}}$$

: Nilai *crisp* yang diperoleh menggunakan metode bisector.

Hasil

$$z_{\text{bisector}}$$

adalah titik di mana area kumulatif di sebelah kiri sama dengan area kumulatif di sebelah kanan.

Aspek–aspek Penting yang Perlu Diperhatikan

1. **Pembagian Seimbang**: Metode bisector memastikan bahwa area di kedua sisi garis vertikal adalah sama.
2. **Tidak Sensitif terhadap Outlier**: Tidak seperti metode centroid, metode bisector kurang terpengaruh oleh ekor panjang dalam fungsi keanggotaan.
3. **Interpretasi**: Nilai bisector merepresentasikan “median” dari distribusi area himpunan fuzzy.
4. **Representasi Seimbang**: Memastikan distribusi area himpunan fuzzy yang setara.
5. **Robustness (Ketahanan)**: Bekerja dengan baik untuk fungsi keanggotaan asimetris atau tidak beraturan.
6. **Interpretasi Fisik**: Merepresentasikan “median” dari distribusi area himpunan fuzzy.
7. **Intensitas Komputasi**: Memerlukan penyelesaian persamaan untuk menemukan titik bisector, yang bisa menjadi menantang untuk fungsi keanggotaan yang kompleks.
8. **Kurang Intuitif**: Mungkin tidak selalu selaras dengan intuisi manusia dibandingkan dengan metode centroid.

Langkah–langkah Menghitung Bisector

1. Definisikan Fungsi Keanggotaan:

- Tentukan semesta pembicaraan (

$$x$$

) dan fungsi keanggotaan (

$$\mu(x)$$

).

2. Hitung Total Area:

- Hitung total area di bawah fungsi keanggotaan:

$$A_{\text{total}} = \int \mu(x) dx$$

3. Cari Titik Bisector:

- Selesaikan persamaan untuk

$$z_{\text{bisector}}$$

sedemikian rupa sehingga:

$$\int_{-\infty}^{z_{\text{bisector}}} \mu(x) dx = \frac{A_{\text{total}}}{2}$$

- Ini melibatkan pencarian nilai

$$x$$

di mana area kumulatif sama dengan setengah dari total area.

4. Keluarkan Hasil:

- Nilai

$$z_{\text{bisector}}$$

adalah *output crisp*.

Kelebihan

1. **Representasi Seimbang:** Memastikan distribusi area himpunan fuzzy yang setara.
2. **Robustness (Ketahanan):** Bekerja dengan baik untuk fungsi keanggotaan asimetris atau tidak beraturan.
3. **Interpretasi Fisik:** Merepresentasikan “median” dari distribusi area himpunan fuzzy.

Kekurangan

1. **Intensitas Komputasi:** Memerlukan penyelesaian persamaan untuk menemukan titik bisector, yang bisa menjadi menantang untuk fungsi keanggotaan yang kompleks.
2. **Kurang Intuitif:** Mungkin tidak selalu selaras dengan intuisi manusia dibandingkan dengan metode centroid.

Aplikasi

Metode bisector umum digunakan dalam: - Sistem kendali fuzzy di mana *output* yang seimbang diperlukan. - Sistem pengambilan keputusan dengan himpunan fuzzy asimetris. - Aplikasi di mana metode centroid dapat menghasilkan hasil yang bias karena bentuk yang tidak beraturan.

Dengan memahami metode bisector, mahasiswa dapat mengapresiasi perannya dalam memberikan nilai *crisp* yang seimbang yang membagi area himpunan fuzzy secara merata. Hal ini

membuatnya sangat berguna untuk menangani fungsi keanggotaan asimetris atau tidak beraturan.

Contoh Proses Matematika Bisector

1. Definisi Fungsi Keanggotaan

Fungsi keanggotaan trapesium asimetris didefinisikan sebagai:

$$\mu_{\text{trap}}(x) = \begin{cases} 0 & \text{jika } x \leq 1, \\ \frac{x-1}{3-1} = \frac{x-1}{2} & \text{jika } 1 < x \leq 3, \\ 1 & \text{jika } 3 < x \leq 7, \\ \frac{11-x}{11-7} = \frac{11-x}{4} & \text{jika } 7 < x \leq 11, \\ 0 & \text{jika } x > 11. \end{cases}$$

2. Rumus Bisector

Nilai bisector

$$z_{\text{bisector}}$$

memenuhi kondisi:

$$\int_{-\infty}^{z_{\text{bisector}}} \mu_{\text{trap}}(x) dx = \int_{z_{\text{bisector}}}^{\infty} \mu_{\text{trap}}(x) dx.$$

Ini berarti area di bawah kurva di sebelah kiri

$$z_{\text{bisector}}$$

sama dengan area di bawah kurva di sebelah kanan

$$z_{\text{bisector}}$$

.

3. Pecah menjadi Wilayah

Fungsi keanggotaan trapesium dapat dibagi menjadi empat wilayah:

1. Wilayah 1:

$$1 < x \leq 3$$

(kenaikan linear).

2. Wilayah 2:

$$3 < x \leq 7$$

(puncak datar).

3. Wilayah 3:

$$7 < x \leq 11$$

(penurunan linear).

4. **Wilayah 4:** Di luar rentang ini,

$$\mu_{\text{trap}}(x) = 0$$

.

4. Hitung Total Area di Bawah Kurva

Pertama, kita hitung total area di bawah fungsi keanggotaan (

$$A_{\text{total}}$$

).

Wilayah 1:

$$1 < x \leq 3$$

$$\begin{aligned} A_1 &= \int_1^3 \mu_{\text{trap}}(x) dx = \int_1^3 \frac{x-1}{2} dx = \frac{1}{2} \int_1^3 (x-1) dx \\ &= \frac{1}{2} \left[\frac{(x-1)^2}{2} \right]_1^3 = \frac{1}{2} \left[\frac{(3-1)^2}{2} - \frac{(1-1)^2}{2} \right] = \frac{1}{2} \cdot \frac{4}{2} = 1. \end{aligned}$$

Wilayah 2:

$$3 < x \leq 7$$

$$A_2 = \int_3^7 \mu_{\text{trap}}(x) dx = \int_3^7 1 dx = [x]_3^7 = 7 - 3 = 4.$$

Wilayah 3:

$$7 < x \leq 11$$

$$\begin{aligned} A_3 &= \int_7^{11} \mu_{\text{trap}}(x) dx = \int_7^{11} \frac{11-x}{4} dx = \frac{1}{4} \int_7^{11} (11-x) dx \\ &= \frac{1}{4} \left[11x - \frac{x^2}{2} \right]_7^{11} = \frac{1}{4} \left[\left(11(11) - \frac{11^2}{2} \right) - \left(11(7) - \frac{7^2}{2} \right) \right] \\ &= \frac{1}{4} \left[\left(121 - \frac{121}{2} \right) - \left(77 - \frac{49}{2} \right) \right] = \frac{1}{4} \left[\frac{121}{2} - \frac{154 - 49}{2} \right] = \frac{1}{4} \left[\frac{121}{2} - \frac{105}{2} \right] = \frac{1}{4} \cdot \frac{16}{2} = 2. \end{aligned}$$

Total Area

$$A_{\text{total}} = A_1 + A_2 + A_3 = 1 + 4 + 2 = 7.$$

Jadi, area di sebelah kiri dan kanan bisector masing-masing harus

$$\frac{A_{\text{total}}}{2} = \frac{7}{2} = 3.5$$

.

5. Cari Nilai Bisector

Sekarang kita cari

$$z_{\text{bisector}}$$

sedemikian rupa sehingga area kumulatif di sebelah kiri sama dengan

$$3.5$$

.

Area Kumulatif di Wilayah 1

$$A_{\text{cumulative},1} = \int_1^3 \mu_{\text{trap}}(x) dx = 1.$$

Karena

$$1 < 3.5$$

, bisector terletak di luar Wilayah 1.

Area Kumulatif di Wilayah 2

Area kumulatif hingga

$$x$$

di Wilayah 2 (

$$3 < x \leq 7$$

) adalah:

$$A_{\text{cumulative},2}(x) = A_1 + \int_3^x \mu_{\text{trap}}(t) dt = 1 + \int_3^x 1 dt = 1 + [t]_3^x = 1 + (x - 3) = x - 2.$$

Kita ingin mencari

$$z_{\text{bisector}}$$

sehingga

$$A_{\text{cumulative},2}(z_{\text{bisector}}) = 3.5$$

.

$$z_{\text{bisector}} - 2 = 3.5$$

Selesaikan untuk

$$z_{\text{bisector}}$$

:

$$z_{\text{bisector}} = 3.5 + 2 = 5.5.$$

Verifikasi: Pastikan nilai

$$z_{\text{bisector}} = 5.5$$

berada dalam Wilayah 2 (

$$3 < x \leq 7$$

). Karena

$$3 < 5.5 \leq 7$$

, maka nilai ini valid.

Area di sebelah kiri

$$z_{\text{bisector}} = 5.5$$

:

$$\int_{-\infty}^{5.5} \mu_{\text{trap}}(x) dx = A_{\text{cumulative},2}(5.5) = 5.5 - 2 = 3.5.$$

Area di sebelah kanan

$$z_{\text{bisector}} = 5.5$$

: Area total adalah 7, dan area di sebelah kiri adalah 3.5, maka area di sebelah kanan juga harus 3.5.

$$\int_{5.5}^{\infty} \mu_{\text{trap}}(x) dx = A_{\text{total}} - \int_{-\infty}^{5.5} \mu_{\text{trap}}(x) dx = 7 - 3.5 = 3.5.$$

Kedua area sama, sehingga

$$z_{\text{bisector}} = 5.5$$

adalah nilai bisector yang benar.

Hasil Akhir

Nilai bisector yang dihitung sesuai dengan keluaran kode program:

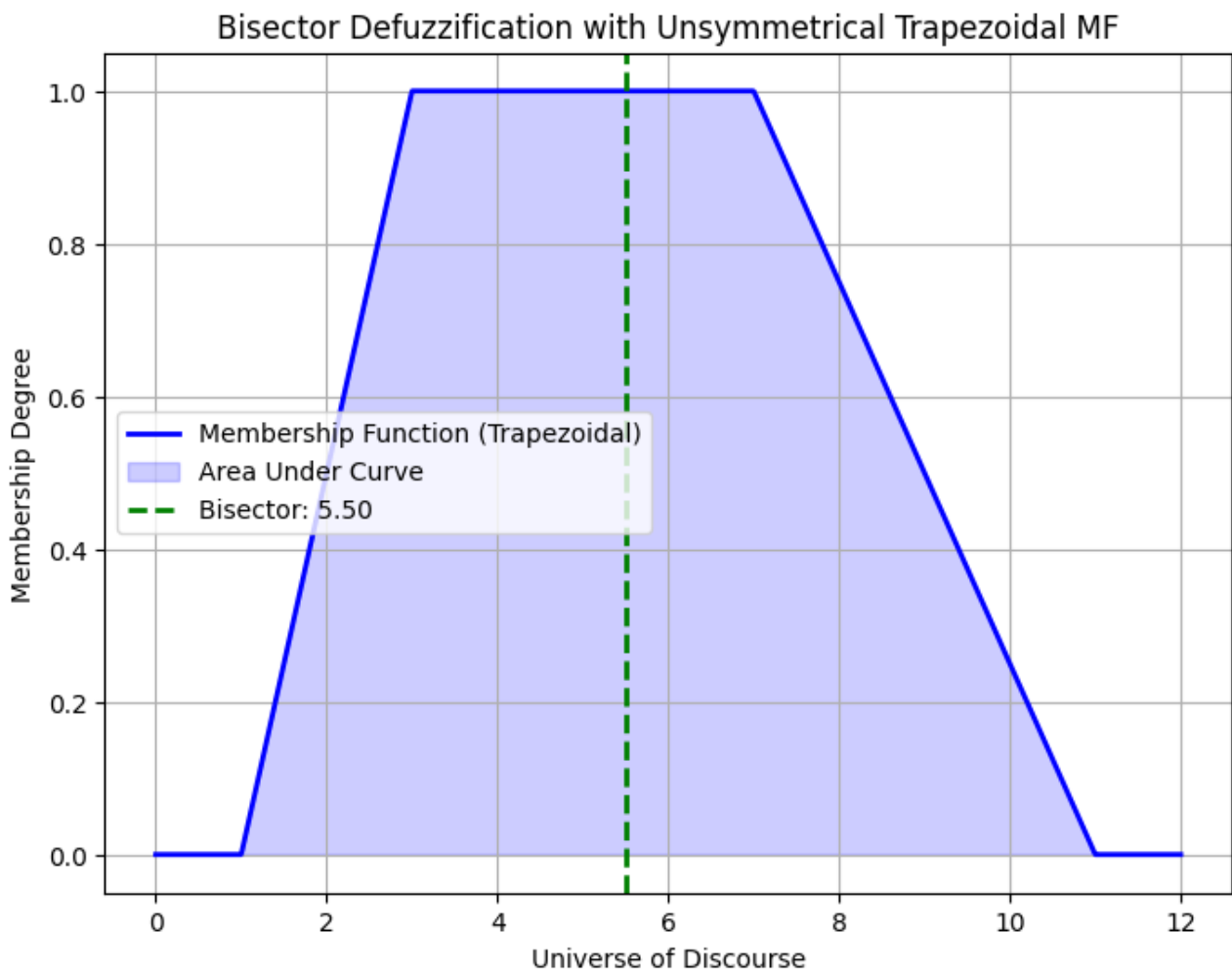
$$z_{\text{bisector}} = 5.50.$$

Contoh Kode Program

Berikut adalah contoh kode program Python yang mengimplementasikan metode bisector untuk defuzifikasi menggunakan fungsi keanggotaan trapesium asimetris.

```
1 import numpy as np
```

```
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
5
6 # Langkah 1: Definisikan semesta pembicaraan
7 x = np.linspace(0, 12, 1000)
8
9 # Langkah 2: Definisikan fungsi keanggotaan trapesium asimetris
10 mfx = fuzz.trapmf(x, [1, 3, 7, 11])
11
12 # Langkah 3: Lakukan defuzifikasi bisector
13 bisector_value = defuzz(x, mfx, mode='bisector')
14
15 # Langkah 4: Visualisasikan fungsi keanggotaan, nilai bisector, dan area di
    bawah kurva
16 plt.figure(figsize=(8, 6))
17
18 # Plot fungsi keanggotaan
19 plt.plot(x, mfx, label='Fungsi Keanggotaan (Trapezium)', color='biru',
    linewidth=2)
20
21 # Isi area di bawah kurva untuk menunjukkan area yang digunakan dalam
    perhitungan bisector
22 plt.fill_between(x, mfx, color='biru', alpha=0.2, label='Area Di Bawah Kurva')
23
24 # Tandai nilai bisector dengan garis vertikal putus-putus
25 plt.axvline(x=bisector_value, color='hijau', linestyle='--', linewidth=2,
    label=f'Bisector: {bisector_value:.2f}')
26
27 # Tambahkan label, judul, legenda, dan grid
28 plt.title('Defuzifikasi Bisector dengan Fungsi Keanggotaan Trapezium Asimetris')
29 plt.xlabel('Semesta Pembicaraan')
30 plt.ylabel('Derajat Keanggotaan')
31 plt.legend()
32 plt.grid(True)
33
34 # Tampilkan plot
35 plt.show()
36
37 # Cetak nilai bisector (opsional)
38 print(f"Nilai Bisector: {bisector_value:.2f}")
```



```
1 Nilai Bisector: 5.50
```

Penjelasan Kode Program

```
1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
```

1. Import Library:

- **numpy**: Digunakan untuk komputasi numerik, seperti mendefinisikan semesta pembicaraan (x) dan melakukan operasi matematika.
- **skfuzzy**: Library untuk operasi logika fuzzy. Menyediakan alat untuk membuat fungsi keanggotaan dan melakukan defuzifikasi.
- **matplotlib.pyplot**: Digunakan untuk memvisualisasikan fungsi keanggotaan dan hasil defuzifikasi.

```
1 # Langkah 3: Lakukan defuzifikasi bisector
2 bisector_value = defuzz(x, mfx, mode='bisector')
```

2. Lakukan Defuzifikasi Bisector:

- Fungsi `defuzz` dari `skfuzzy.defuzzify` digunakan untuk menghitung nilai bisector.
- `mode= 'bisector'` menentukan bahwa metode bisector digunakan.
- Metode bisector mencari nilai

$$x$$

(

$$z_{\text{textbisector}}$$

) sedemikian rupa sehingga area di bawah fungsi keanggotaan di sebelah kiri

$$z_{\text{textbisector}}$$

sama dengan area di sebelah kanan:

$$\int_{-\infty}^{z_{\text{textbisector}}} \mu(x) dx = \int_{z_{\text{textbisector}}}^{\infty} \mu(x) dx$$

- Baris kode ini adalah inti dari proses defuzifikasi bisector dalam program ini. Fungsi `defuzz` mengambil semesta pembicaraan (`x`), fungsi keanggotaan (`mfx`), dan metode defuzifikasi ('bisector') sebagai *input*, dan menghasilkan nilai *crisp* bisector.

OK, berikut adalah draft bagian *handout* mengenai Metode Mean of Maximum (MoM) dalam bahasa Indonesia, mengikuti format yang telah digunakan sebelumnya:

3. Metode Mean of Maximum (MOM) (Rata-rata Maksimum)

Metode Mean of Maximum (MOM) adalah teknik defuzifikasi yang digunakan dalam sistem logika fuzzy untuk menghitung nilai *crisp* dari himpunan fuzzy. Metode ini menghitung rata-rata dari semua nilai

$$x$$

di mana fungsi keanggotaan mencapai nilai maksimumnya. Metode ini sangat berguna ketika himpunan fuzzy memiliki wilayah datar (misalnya, fungsi keanggotaan trapesium atau segitiga dengan *plateau*).

Konsep Dasar

Metode MOM mengidentifikasi wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{\max}$$

) dan menghitung rata-rata dari semua nilai

$$x$$

dalam wilayah tersebut. Secara matematis, ini dinyatakan sebagai:

$$z_{\text{mom}} = \frac{\int x \cdot \delta(\mu(x) - \mu_{\text{max}}) dx}{\int \delta(\mu(x) - \mu_{\text{max}}) dx}$$

Dimana: -

$$x$$

: Semesta pembicaraan (rentang input). -

$$\mu(x)$$

: Fungsi keanggotaan. -

$$\mu_{\text{max}}$$

: Derajat keanggotaan maksimum (biasanya 1 untuk fungsi keanggotaan yang dinormalisasi). -

$$\delta(\mu(x) - \mu_{\text{max}})$$

: Fungsi delta Dirac yang hanya memilih titik-titik di mana

$$\mu(x) = \mu_{\text{max}}$$

.

Untuk kasus diskrit, rumusnya disederhanakan menjadi:

$$z_{\text{mom}} = \frac{\sum x_i}{N}$$

Dimana: -

$$x_i$$

: Semua nilai

$$x$$

di mana

$$\mu(x) = \mu_{\text{max}}$$

. -

$$N$$

: Jumlah total nilai

$$x_i$$

tersebut.

Aspek-aspek Penting yang Perlu Diperhatikan

1. **Fokus pada Wilayah Maksimum:** Metode MOM hanya berfokus pada wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya.

2. **Komputasi Sederhana:** Untuk fungsi keanggotaan dengan puncak datar (misalnya, trapezium), perhitungan menjadi sangat mudah.
3. **Tidak Sensitif terhadap Wilayah Non-Maksimum:** Mengabaikan wilayah fungsi keanggotaan di luar wilayah maksimum.
4. **Kesederhanaan:** Mudah dihitung, terutama untuk fungsi keanggotaan dengan puncak datar.
5. **Interpretasi:** Merepresentasikan “pusat” dari wilayah maksimum.
6. **Ruang Lingkup Terbatas:** Hanya mempertimbangkan wilayah maksimum, mengabaikan bagian lain dari fungsi keanggotaan.
7. **Tidak Sensitif terhadap Bentuk:** Tidak memperhitungkan bentuk atau distribusi himpunan fuzzy di luar wilayah maksimum.

Langkah-langkah Menghitung MOM

1. Definisikan Fungsi Keanggotaan:

- Tentukan semesta pembicaraan (x) dan fungsi keanggotaan ($\mu(x)$).

2. Identifikasi Derajat Keanggotaan Maksimum:

- Temukan derajat keanggotaan maksimum (μ_{\max}) dari fungsi keanggotaan.

3. Temukan Wilayah di Mana

$$\mu(x) = \mu_{\max}$$

:

- Identifikasi semua nilai x di mana fungsi keanggotaan sama dengan μ_{\max} .

4. Hitung Rata-rata:

- Hitung rata-rata dari semua nilai x

di wilayah maksimum:

$$z_{\text{mom}} = \frac{\sum x_i}{N}$$

5. Keluarkan Hasil:

- Nilai

$$z_{\text{mom}}$$

adalah *output crisp*.

Kelebihan

1. **Kesederhanaan:** Mudah dihitung, terutama untuk fungsi keanggotaan dengan puncak datar.
2. **Fokus pada Wilayah Maksimum:** Mengabaikan bagian-bagian fungsi keanggotaan yang tidak relevan.
3. **Interpretasi:** Merepresentasikan “pusat” dari wilayah maksimum.

Kekurangan

1. **Ruang Lingkup Terbatas:** Hanya mempertimbangkan wilayah maksimum, mengabaikan bagian lain dari fungsi keanggotaan.
2. **Tidak Sensitif terhadap Bentuk:** Tidak memperhitungkan bentuk atau distribusi himpunan fuzzy di luar wilayah maksimum.

Aplikasi

Metode MOM umum digunakan dalam: – Sistem kendali fuzzy di mana fokusnya adalah pada wilayah himpunan fuzzy yang paling pasti. – Sistem pengambilan keputusan dengan fungsi keanggotaan yang memiliki puncak datar. – Aplikasi di mana kesederhanaan dan efisiensi komputasi diprioritaskan.

Dengan memahami metode MOM, mahasiswa dapat mengapresiasi perannya dalam memberikan nilai *crisp* yang merepresentasikan tendensi sentral dari wilayah maksimum himpunan fuzzy. Hal ini membuatnya sangat berguna untuk menangani fungsi keanggotaan dengan puncak datar atau *plateau*.

Contoh Perhitungan Matematika Mean of Maximum (MOM)**1. Definisi Fungsi Keanggotaan**

Fungsi keanggotaan trapesium asimetris didefinisikan sebagai:

$$\mu_{\text{trap}}(x) = \begin{cases} 0 & \text{jika } x \leq 1, \\ \frac{x-1}{3-1} = \frac{x-1}{2} & \text{jika } 1 < x \leq 3, \\ 1 & \text{jika } 3 < x \leq 7, \\ \frac{11-x}{11-7} = \frac{11-x}{4} & \text{jika } 7 < x \leq 11, \\ 0 & \text{jika } x > 11. \end{cases}$$

2. Identifikasi Derajat Keanggotaan Maksimum

Derajat keanggotaan maksimum (

$$\mu_{\max}$$

) dari fungsi keanggotaan trapesium adalah:

$$\mu_{\max} = 1.$$

3. Temukan Wilayah di Mana

$$\mu(x) = \mu_{\max}$$

Fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{\max} = 1$$

) di wilayah puncak datar:

$$3 \leq x \leq 7.$$

Dengan demikian, himpunan nilai

$$x$$

di mana

$$\mu(x) = \mu_{\max}$$

adalah:

$$x \in [3, 7].$$

4. Hitung Mean of Maximum

Mean of Maximum (MOM) adalah rata-rata dari semua nilai

$$x$$

dalam wilayah di mana

$$\mu(x) = \mu_{\max}$$

. Untuk rentang kontinu

$$[a, b]$$

, rata-rata diberikan oleh:

$$z_{\text{mom}} = \frac{a + b}{2}.$$

Substitusikan

$$a = 3$$

dan

$$b = 7$$

:

$$z_{\text{mom}} = \frac{3 + 7}{2} = \frac{10}{2} = 5.$$

Hasil Akhir

Nilai Mean of Maximum yang dihitung sesuai dengan keluaran kode program:

$$z_{\text{mom}} = 5.00.$$

Contoh Kode Program

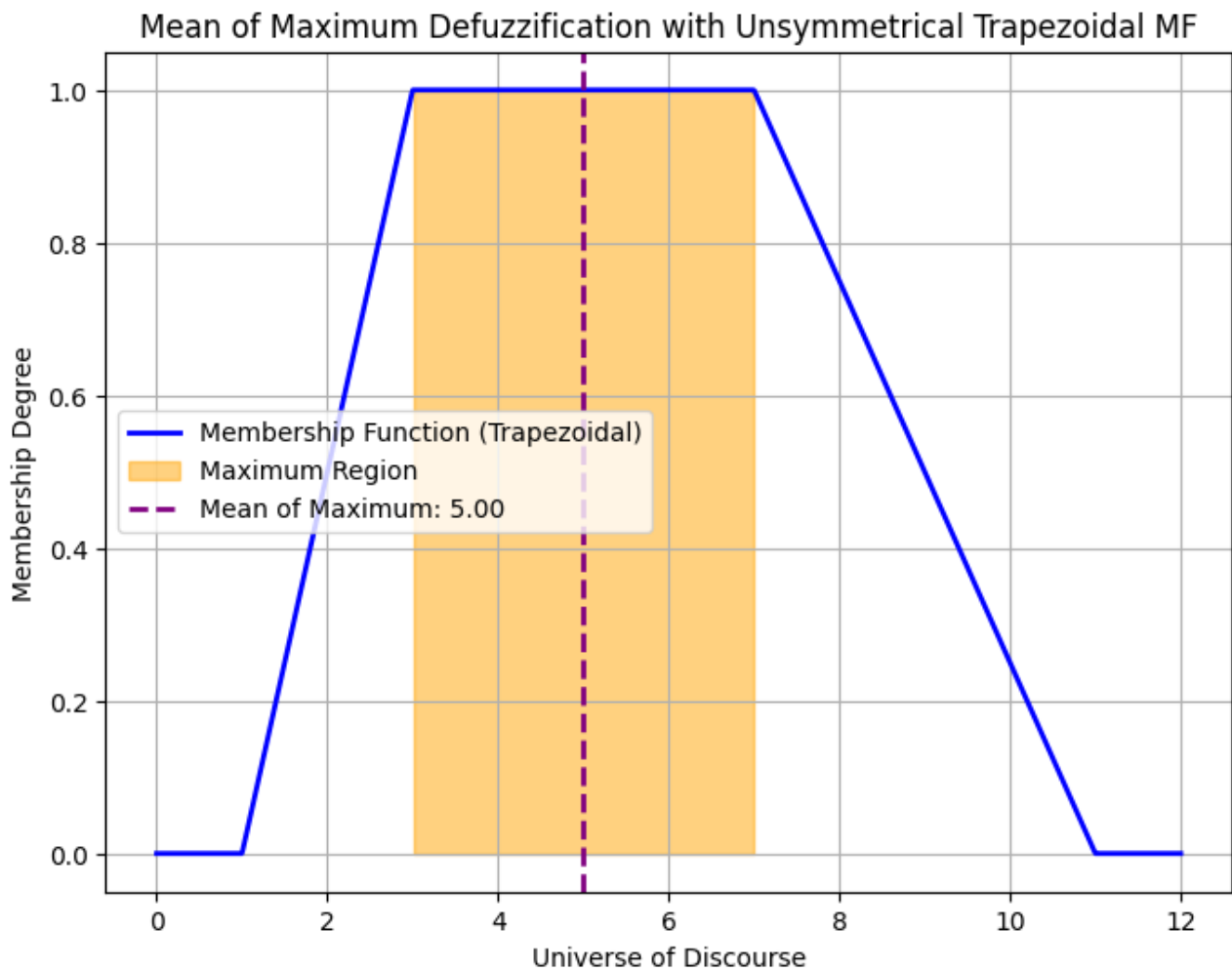
Berikut adalah contoh kode program Python yang mengimplementasikan metode Mean of Maximum (MOM) untuk defuzifikasi menggunakan fungsi keanggotaan trapesium asimetris.

```
1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
5
6 # Langkah 1: Definisikan semesta pembicaraan
7 x = np.linspace(0, 12, 1000)
8
9 # Langkah 2: Definisikan fungsi keanggotaan trapesium asimetris
10 mfx = fuzz.trapmf(x, [1, 3, 7, 11])
11
12 # Langkah 3: Lakukan defuzifikasi Mean of Maximum (MOM)
13 mom_value = defuzz(x, mfx, mode='mom')
14
15 # Langkah 4: Visualisasikan fungsi keanggotaan dan nilai MOM
16 plt.figure(figsize=(8, 6))
17
18 # Plot fungsi keanggotaan
19 plt.plot(x, mfx, label='Fungsi Keanggotaan (Trapezium)', color='biru',
20         linewidth=2)
21
22 # Sorot wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya
23 max_membership = np.max(mfx)
24 plt.fill_between(x, mfx, where=(mfx == max_membership), color='orange', alpha
25                 =0.5, label='Wilayah Maksimum')
26
27 # Tandai nilai MOM dengan garis vertikal putus-putus
28 plt.axvline(x=mom_value, color='ungu', linestyle='--', linewidth=2, label=f'
29             Mean of Maximum: {mom_value:.2f}')
30
31 # Tambahkan label, judul, legenda, dan grid
```

```

29 plt.title('Defuzifikasi Mean of Maximum dengan Fungsi Keanggotaan Trapesium
    Asimetris')
30 plt.xlabel('Semesta Pembicaraan')
31 plt.ylabel('Derajat Keanggotaan')
32 plt.legend()
33 plt.grid(True)
34
35 # Tampilkan plot
36 plt.show()
37
38 # Cetak nilai MOM (opsional)
39 print(f"Nilai Mean of Maximum: {mom_value:.2f}")

```



```

1 Nilai Mean of Maximum: 5.00

```

Penjelasan Kode Program

```

1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt

```

1. Import Library:

- **numpy**: Digunakan untuk komputasi numerik, seperti mendefinisikan semesta pem-

bicaraan (x) dan melakukan operasi matematika.

- **skfuzzy**: Library untuk operasi logika fuzzy. Menyediakan alat untuk membuat fungsi keanggotaan dan melakukan defuzifikasi.
- **matplotlib.pyplot**: Digunakan untuk memvisualisasikan fungsi keanggotaan dan hasil defuzifikasi.

```
1 # Langkah 3: Lakukan defuzifikasi Mean of Maximum (MOM)
2 mom_value = defuzz(x, mfx, mode='mom')
```

2. Lakukan Defuzifikasi Mean of Maximum (MOM):

- Fungsi **defuzz** dari **skfuzzy.defuzzify** digunakan untuk menghitung nilai MOM.
- **mode= 'mom'** menentukan bahwa metode Mean of Maximum digunakan.
- Metode MOM menghitung rata-rata dari semua nilai

$$x$$

di mana fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{\text{textmax}}$$

).

- Baris kode ini adalah inti dari proses defuzifikasi MOM dalam program ini. Fungsi **defuzz** mengambil semesta pembicaraan (x), fungsi keanggotaan (**mfx**), dan metode defuzifikasi ('mom') sebagai *input*, dan menghasilkan nilai *crisp* MOM.

Baik, berikut adalah draft bagian *handout* mengenai Metode Smallest of Maximum (SoM) dalam bahasa Indonesia, dengan format yang serupa dengan bagian-bagian sebelumnya:

4. Metode Smallest of Maximum (SoM) (Nilai Maksimum Terkecil)

Metode Smallest of Maximum (SoM) adalah teknik defuzifikasi yang digunakan dalam sistem logika fuzzy untuk menghitung nilai *crisp* dari himpunan fuzzy. Metode ini memilih nilai

$$x$$

terkecil di mana fungsi keanggotaan mencapai nilai maksimumnya. Metode ini sangat berguna ketika himpunan fuzzy memiliki banyak titik atau wilayah dengan derajat keanggotaan maksimum yang sama.

Konsep Dasar

Metode SoM mengidentifikasi wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{\text{max}}$$

) dan memilih nilai

$$x$$

terkecil dalam wilayah tersebut. Secara matematis, ini dinyatakan sebagai:

$$z_{\text{som}} = \min(x_i)$$

Dimana: -

$$x_i$$

: Semua nilai

$$x$$

di mana fungsi keanggotaan sama dengan

$$\mu_{\text{max}}$$

. -

$$\mu_{\text{max}}$$

: Derajat keanggotaan maksimum (biasanya 1 untuk fungsi keanggotaan yang dinormalisasi).

Untuk fungsi keanggotaan kontinu, ini sesuai dengan titik paling kiri dari wilayah maksimum.

Aspek-aspek Penting yang Perlu Diperhatikan

1. **Fokus pada Wilayah Maksimum:** Metode SoM hanya berfokus pada wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya.
2. **Memilih Nilai Terkecil:** Di antara semua nilai

$$x$$

dalam wilayah maksimum, nilai terkecil yang dipilih.

3. **Tidak Sensitif terhadap Wilayah Non-Maksimum:** Mengabaikan wilayah fungsi keanggotaan di luar wilayah maksimum.
4. **Kesederhanaan:** Mudah dihitung, terutama untuk fungsi keanggotaan dengan puncak datar.
5. **Pilihan Konservatif:** Memilih nilai

$$x$$

terkecil, yang bisa berguna dalam aplikasi di mana nilai yang lebih rendah lebih disukai.

6. **Interpretasi:** Merepresentasikan nilai terkecil dari wilayah dengan keanggotaan maksimum.
7. **Ruang Lingkup Terbatas:** Hanya mempertimbangkan wilayah maksimum, mengabaikan bagian lain dari fungsi keanggotaan.
8. **Tidak Sensitif terhadap Bentuk:** Tidak memperhitungkan bentuk atau distribusi himpunan fuzzy di luar wilayah maksimum.
9. **Bias terhadap Nilai Lebih Kecil:** Mungkin tidak selalu selaras dengan intuisi manusia jika nilai yang lebih besar lebih diinginkan.

Langkah–langkah Menghitung SoM

1. Definisikan Fungsi Keanggotaan:

- Tentukan semesta pembicaraan (x) dan fungsi keanggotaan ($\mu(x)$).

2. Identifikasi Derajat Keanggotaan Maksimum:

- Temukan derajat keanggotaan maksimum (μ_{\max}) dari fungsi keanggotaan.

3. Temukan Wilayah di Mana

$$\mu(x) = \mu_{\max}$$

:

- Identifikasi semua nilai x di mana fungsi keanggotaan sama dengan μ_{\max} .

4. Pilih Nilai

$$x$$

Terkecil:

- Dari nilai x yang teridentifikasi, pilih yang terkecil:

$$z_{\text{som}} = \min(x_i)$$

5. Keluarkan Hasil:

- Nilai z_{som} adalah *output crisp*.

Kelebihan

1. **Kesederhanaan:** Mudah dihitung, terutama untuk fungsi keanggotaan dengan puncak datar.
2. **Pilihan Konservatif:** Memilih nilai

$$x$$

terkecil, yang bisa berguna dalam aplikasi di mana nilai yang lebih rendah lebih disukai.

3. **Fokus pada Wilayah Maksimum:** Mengabaikan bagian-bagian fungsi keanggotaan yang tidak relevan.

Kekurangan

1. **Ruang Lingkup Terbatas:** Hanya mempertimbangkan wilayah maksimum, mengabaikan bagian lain dari fungsi keanggotaan.
2. **Tidak Sensitif terhadap Bentuk:** Tidak memperhitungkan bentuk atau distribusi himpunan fuzzy di luar wilayah maksimum.
3. **Bias terhadap Nilai Lebih Kecil:** Mungkin tidak selalu selaras dengan intuisi manusia jika nilai yang lebih besar lebih diinginkan.

Aplikasi

Metode SoM umum digunakan dalam: – Sistem kendali fuzzy di mana keputusan konservatif lebih disukai. – Sistem pengambilan keputusan dengan fungsi keanggotaan yang memiliki puncak datar. – Aplikasi di mana kesederhanaan dan efisiensi komputasi diprioritaskan.

Dengan memahami metode SoM, mahasiswa dapat mengapresiasi perannya dalam memberikan nilai *crisp* yang merepresentasikan nilai

$$x$$

terkecil dalam wilayah maksimum himpunan fuzzy. Hal ini membuatnya sangat berguna untuk menangani fungsi keanggotaan dengan puncak datar atau *plateau*, terutama ketika nilai yang lebih kecil lebih disukai.

Contoh Perhitungan Matematika Smallest of Maximum (SoM)

1. Definisi Fungsi Keanggotaan

Fungsi keanggotaan trapesium asimetris didefinisikan sebagai:

$$\mu_{\text{trap}}(x) = \begin{cases} 0 & \text{jika } x \leq 1, \\ \frac{x-1}{3-1} = \frac{x-1}{2} & \text{jika } 1 < x \leq 3, \\ 1 & \text{jika } 3 \leq x \leq 7, \\ \frac{11-x}{11-7} = \frac{11-x}{4} & \text{jika } 7 < x \leq 11, \\ 0 & \text{jika } x > 11. \end{cases}$$

2. Identifikasi Derajat Keanggotaan Maksimum

Derajat keanggotaan maksimum (

$$\mu_{\max}$$

) dari fungsi keanggotaan trapesium adalah:

$$\mu_{\max} = 1.$$

3. Temukan Wilayah di Mana

$$\mu(x) = \mu_{\max}$$

Fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{\max} = 1$$

) di wilayah puncak datar:

$$3 \leq x \leq 7.$$

Dengan demikian, himpunan nilai

$$x$$

di mana

$$\mu(x) = \mu_{\max}$$

adalah interval tertutup:

$$x \in [3, 7].$$

4. Hitung Smallest of Maximum (SoM)

Smallest of Maximum (SoM) adalah nilai

$$x$$

terkecil dalam wilayah di mana

$$\mu(x) = \mu_{\max}$$

. Untuk rentang

$$[3, 7]$$

, nilai

$$x$$

terkecil adalah batas bawah dari interval ini:

$$z_{\text{som}} = \min\{x \mid 3 \leq x \leq 7\} = 3.$$

Dalam interval

$$[3, 7]$$

, nilai terkecil dari

$$x$$

adalah 3.

Hasil Akhir

Nilai Smallest of Maximum yang dihitung sesuai dengan keluaran kode program:

$$z_{\text{som}} = 3.00.$$

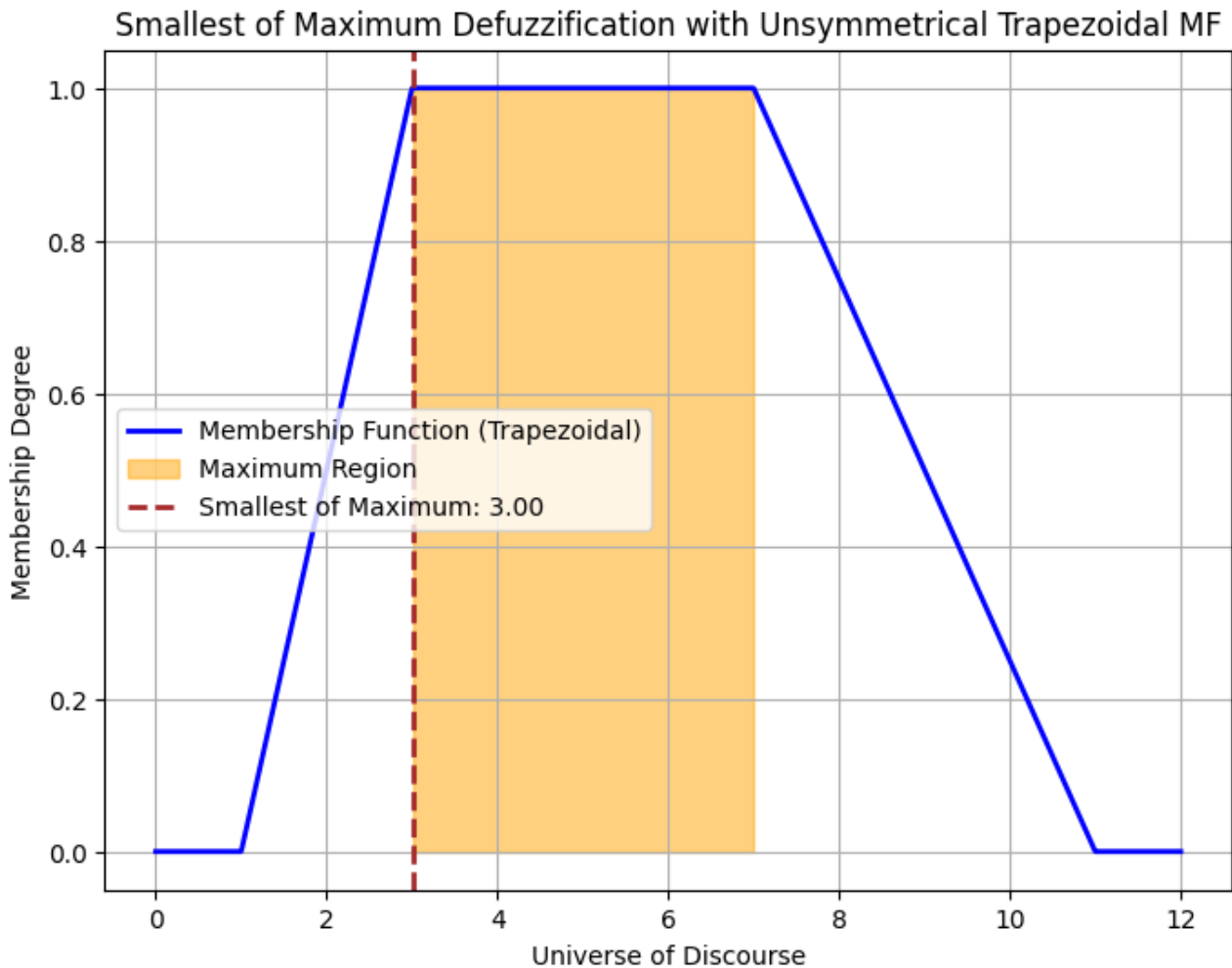
Contoh Kode Program

Berikut adalah contoh kode program Python yang mengimplementasikan metode Smallest of Maximum (SoM) untuk defuzifikasi menggunakan fungsi keanggotaan trapesium asimetris.

```

1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
5
6 # Langkah 1: Definisikan semesta pembicaraan
7 x = np.linspace(0, 12, 1000)
8
9 # Langkah 2: Definisikan fungsi keanggotaan trapesium asimetris
10 mfx = fuzz.trapmf(x, [1, 3, 7, 11])
11
12 # Langkah 3: Lakukan defuzifikasi Smallest of Maximum (SoM)
13 som_value = defuzz(x, mfx, mode='som')
14
15 # Langkah 4: Visualisasikan fungsi keanggotaan dan nilai SoM
16 plt.figure(figsize=(8, 6))
17
18 # Plot fungsi keanggotaan
19 plt.plot(x, mfx, label='Fungsi Keanggotaan (Trapezium)', color='biru',
20          linewidth=2)
21
22 # Sorot wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya
23 max_membership = np.max(mfx)
24 plt.fill_between(x, mfx, where=(mfx == max_membership), color='orange', alpha
25                  =0.5, label='Wilayah Maksimum')
26
27 # Tandai nilai SoM dengan garis vertikal putus-putus
28 plt.axvline(x=som_value, color='coklat', linestyle='--', linewidth=2, label=f'
29             Smallest of Maximum: {som_value:.2f}')
30
31 # Tambahkan label, judul, legenda, dan grid
32 plt.title('Defuzifikasi Smallest of Maximum dengan Fungsi Keanggotaan
33           Trapezium Asimetris')
34 plt.xlabel('Semesta Pembicaraan')
35 plt.ylabel('Derajat Keanggotaan')
36 plt.legend()
37 plt.grid(True)
38
39 # Tampilkan plot
40 plt.show()
41
42 # Cetak nilai SoM (opsional)
43 print(f"Nilai Smallest of Maximum: {som_value:.2f}")

```



```
1 Nilai Smallest of Maximum: 3.00
```

Penjelasan Kode Program

```
1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
```

1. Import Library:

- **numpy**: Digunakan untuk komputasi numerik, seperti mendefinisikan semesta pembicaraan (x) dan melakukan operasi matematika.
- **skfuzzy**: Library untuk operasi logika fuzzy. Menyediakan alat untuk membuat fungsi keanggotaan dan melakukan defuzzifikasi.
- **matplotlib.pyplot**: Digunakan untuk memvisualisasikan fungsi keanggotaan dan hasil defuzzifikasi.

```
1 # Langkah 3: Lakukan defuzzifikasi Smallest of Maximum (SoM)
2 som_value = defuzz(x, mfx, mode='som')
```

2. Lakukan Defuzzifikasi Smallest of Maximum (SoM):

- Fungsi `defuzz` dari `skfuzzy.defuzzify` digunakan untuk menghitung nilai SoM.
- `mode= 'som'` menentukan bahwa metode Smallest of Maximum digunakan.
- Metode SoM memilih nilai

$$x$$

terkecil di mana fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{textmax}$$

).

- Baris kode ini adalah inti dari proses defuzifikasi SoM dalam program ini. Fungsi `defuzz` mengambil semesta pembicaraan (x), fungsi keanggotaan ($\mu_f x$), dan metode defuzifikasi ('som') sebagai *input*, dan menghasilkan nilai *crisp* SoM.

Baik, berikut adalah draft bagian *handout* mengenai Metode Largest of Maximum (LoM) dalam bahasa Indonesia, mengikuti format yang telah digunakan untuk metode-metode defuzifikasi sebelumnya:

5. Metode Largest of Maximum (LoM) (Nilai Maksimum Terbesar)

Metode Largest of Maximum (LoM) adalah teknik defuzifikasi yang digunakan dalam sistem logika fuzzy untuk menghitung nilai *crisp* dari himpunan fuzzy. Metode ini memilih nilai

$$x$$

terbesar di mana fungsi keanggotaan mencapai nilai maksimumnya. Metode ini sangat berguna ketika himpunan fuzzy memiliki banyak titik atau wilayah dengan derajat keanggotaan maksimum yang sama.

Konsep Dasar

Metode LoM mengidentifikasi wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{\max}$$

) dan memilih nilai

$$x$$

terbesar dalam wilayah tersebut. Secara matematis, ini dinyatakan sebagai:

$$z_{\text{lom}} = \max(x_i)$$

Dimana: -

$$x_i$$

: Semua nilai

$$x$$

di mana fungsi keanggotaan sama dengan

$$\mu_{\max}$$

.-

$$\mu_{\max}$$

: Derajat keanggotaan maksimum (biasanya 1 untuk fungsi keanggotaan yang dinormalisasi).

Untuk fungsi keanggotaan kontinu, ini sesuai dengan titik paling kanan dari wilayah maksimum.

Aspek-aspek Penting yang Perlu Diperhatikan

1. **Fokus pada Wilayah Maksimum:** Metode LoM hanya berfokus pada wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya.
2. **Memilih Nilai Terbesar:** Di antara semua nilai

$$x$$

dalam wilayah maksimum, nilai terbesar yang dipilih.

3. **Tidak Sensitif terhadap Wilayah Non-Maksimum:** Mengabaikan wilayah fungsi keanggotaan di luar wilayah maksimum.
4. **Kesederhanaan:** Mudah dihitung, terutama untuk fungsi keanggotaan dengan puncak datar.
5. **Pilihan Agresif:** Memilih nilai

$$x$$

terbesar, yang bisa berguna dalam aplikasi di mana nilai yang lebih tinggi lebih disukai.

6. **Interpretasi:** Merepresentasikan nilai terbesar dari wilayah dengan keanggotaan maksimum.
7. **Ruang Lingkup Terbatas:** Hanya mempertimbangkan wilayah maksimum, mengabaikan bagian lain dari fungsi keanggotaan.
8. **Tidak Sensitif terhadap Bentuk:** Tidak memperhitungkan bentuk atau distribusi himpunan fuzzy di luar wilayah maksimum.
9. **Bias terhadap Nilai Lebih Besar:** Mungkin tidak selalu selaras dengan intuisi manusia jika nilai yang lebih kecil lebih diinginkan.

Langkah-langkah Menghitung LoM

1. **Definisikan Fungsi Keanggotaan:**

- Tentukan semesta pembicaraan (

$$x$$

) dan fungsi keanggotaan (

$$\mu(x)$$

).

2. Identifikasi Derajat Keanggotaan Maksimum:

- Temukan derajat keanggotaan maksimum (

$$\mu_{\max}$$

) dari fungsi keanggotaan.

3. Temukan Wilayah di Mana

$$\mu(x) = \mu_{\max}$$

:

- Identifikasi semua nilai

$$x$$

di mana fungsi keanggotaan sama dengan

$$\mu_{\max}$$

.

4. Pilih Nilai

$$x$$

Terbesar:

- Dari nilai

$$x$$

yang teridentifikasi, pilih yang terbesar:

$$z_{\text{lom}} = \max(x_i)$$

5. Keluarkan Hasil:

- Nilai

$$z_{\text{lom}}$$

adalah *output crisp*.

Kelebihan

1. **Kesederhanaan:** Mudah dihitung, terutama untuk fungsi keanggotaan dengan puncak datar.
2. **Pilihan Agresif:** Memilih nilai
$$x$$

terbesar, yang bisa berguna dalam aplikasi di mana nilai yang lebih tinggi lebih disukai.
3. **Fokus pada Wilayah Maksimum:** Mengabaikan bagian-bagian fungsi keanggotaan yang tidak relevan.

Kekurangan

1. **Ruang Lingkup Terbatas:** Hanya mempertimbangkan wilayah maksimum, mengabaikan bagian lain dari fungsi keanggotaan.
2. **Tidak Sensitif terhadap Bentuk:** Tidak memperhitungkan bentuk atau distribusi himpunan fuzzy di luar wilayah maksimum.
3. **Bias terhadap Nilai Lebih Besar:** Mungkin tidak selalu selaras dengan intuisi manusia jika nilai yang lebih kecil lebih diinginkan.

Aplikasi

Metode LoM umum digunakan dalam: – Sistem kendali fuzzy di mana keputusan agresif lebih disukai. – Sistem pengambilan keputusan dengan fungsi keanggotaan yang memiliki puncak datar. – Aplikasi di mana kesederhanaan dan efisiensi komputasi diprioritaskan.

Dengan memahami metode LoM, mahasiswa dapat mengapresiasi perannya dalam memberikan nilai *crisp* yang merepresentasikan nilai

$$x$$

terbesar dalam wilayah maksimum himpunan fuzzy. Hal ini membuatnya sangat berguna untuk menangani fungsi keanggotaan dengan puncak datar atau *plateau*, terutama ketika nilai yang lebih besar lebih disukai.

Contoh Perhitungan Matematika Largest of Maximum (LoM)

1. Definisi Fungsi Keanggotaan

Fungsi keanggotaan trapesium asimetris didefinisikan sebagai:

$$\mu_{\text{trap}}(x) = \begin{cases} 0 & \text{jika } x \leq 1, \\ \frac{x-1}{3-1} = \frac{x-1}{2} & \text{jika } 1 < x \leq 3, \\ 1 & \text{jika } 3 \leq x \leq 7, \\ \frac{11-x}{11-7} = \frac{11-x}{4} & \text{jika } 7 < x \leq 11, \\ 0 & \text{jika } x > 11. \end{cases}$$

2. Identifikasi Derajat Keanggotaan Maksimum

Derajat keanggotaan maksimum (

$$\mu_{\text{max}}$$

) dari fungsi keanggotaan trapesium adalah:

$$\mu_{\text{max}} = 1.$$

3. Temukan Wilayah di Mana

$$\mu(x) = \mu_{\text{max}}$$

Fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{\max} = 1$$

) di wilayah puncak datar:

$$3 \leq x \leq 7.$$

Dengan demikian, himpunan nilai

$$x$$

di mana

$$\mu(x) = \mu_{\max}$$

adalah interval tertutup:

$$x \in [3, 7].$$

4. Hitung Largest of Maximum (LoM)

Largest of Maximum (LoM) adalah nilai

$$x$$

terbesar dalam wilayah di mana

$$\mu(x) = \mu_{\max}$$

. Untuk rentang

$$[3, 7]$$

, nilai

$$x$$

terbesar adalah batas atas dari interval ini:

$$z_{\text{lom}} = \max\{x \mid 3 \leq x \leq 7\} = 7.$$

Dalam interval

$$[3, 7]$$

, nilai terbesar dari

$$x$$

adalah 7.

Hasil Akhir

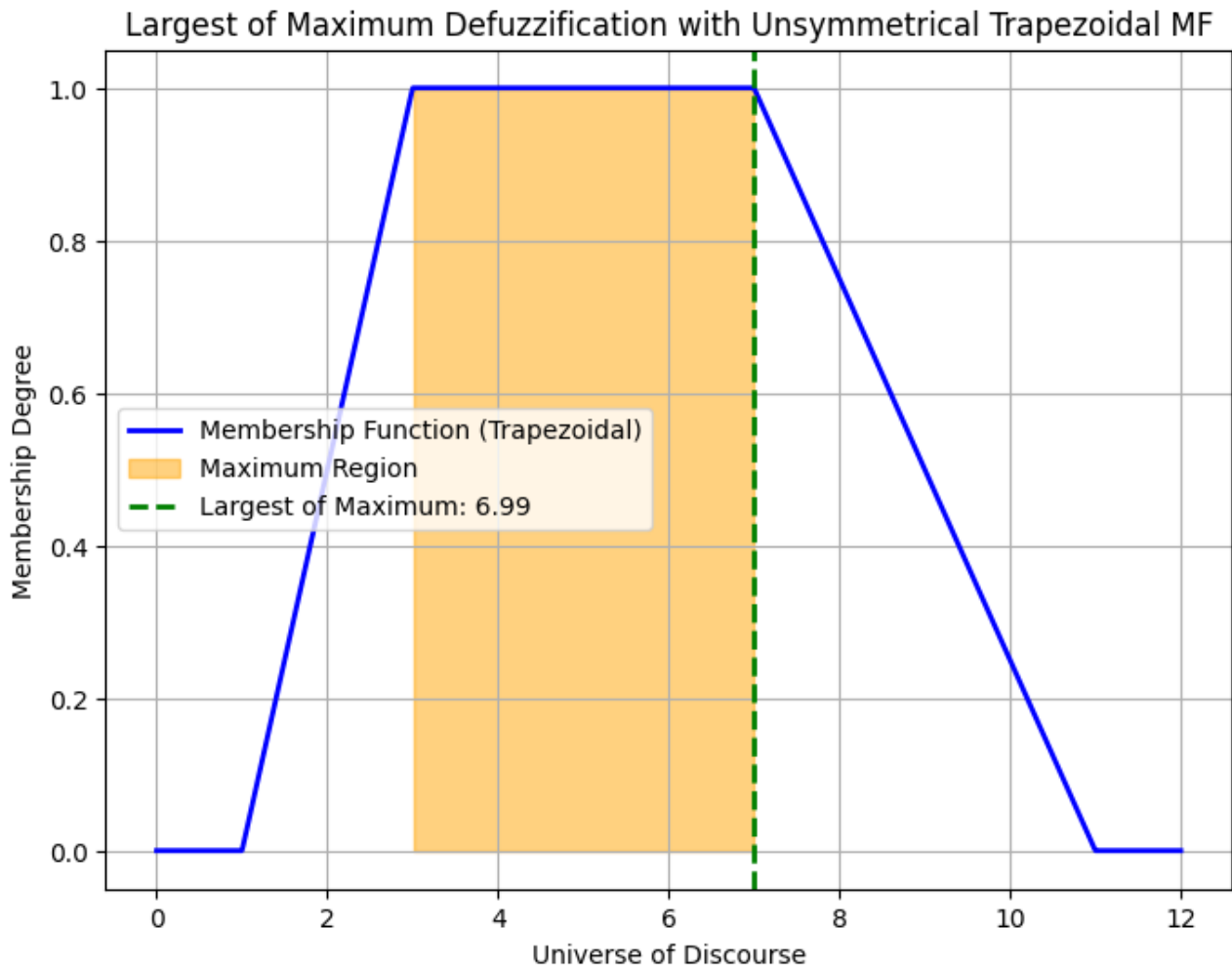
Nilai Largest of Maximum yang dihitung sesuai dengan keluaran kode program:

$$z_{\text{lom}} = 7.00.$$

Contoh Kode Program

Berikut adalah contoh kode program Python yang mengimplementasikan metode Largest of Maximum (LoM) untuk defuzifikasi menggunakan fungsi keanggotaan trapesium asimetris.

```
1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
5
6 # Langkah 1: Definisikan semesta pembicaraan
7 x = np.linspace(0, 12, 1000)
8
9 # Langkah 2: Definisikan fungsi keanggotaan trapesium asimetris
10 mfx = fuzz.trapmf(x, [1, 3, 7, 11])
11
12 # Langkah 3: Lakukan defuzifikasi Largest of Maximum (LoM)
13 lom_value = defuzz(x, mfx, mode='lom')
14
15 # Langkah 4: Visualisasikan fungsi keanggotaan dan nilai LoM
16 plt.figure(figsize=(8, 6))
17
18 # Plot fungsi keanggotaan
19 plt.plot(x, mfx, label='Fungsi Keanggotaan (Trapezium)', color='biru',
20         linewidth=2)
21
22 # Sorot wilayah di mana fungsi keanggotaan mencapai nilai maksimumnya
23 max_membership = np.max(mfx)
24 plt.fill_between(x, mfx, where=(mfx == max_membership), color='orange', alpha
25                 =0.5, label='Wilayah Maksimum')
26
27 # Tandai nilai LoM dengan garis vertikal putus-putus
28 plt.axvline(x=lom_value, color='hijau', linestyle='--', linewidth=2, label=f'
29         Largest of Maximum: {lom_value:.2f}')
30
31 # Tambahkan label, judul, legenda, dan grid
32 plt.title('Defuzifikasi Largest of Maximum dengan Fungsi Keanggotaan Trapezium
33         Asimetris')
34 plt.xlabel('Semesta Pembicaraan')
35 plt.ylabel('Derajat Keanggotaan')
36 plt.legend()
37 plt.grid(True)
38
39 # Tampilkan plot
40 plt.show()
41
42 # Cetak nilai LoM (opsional)
43 print(f"Nilai Largest of Maximum: {lom_value:.2f}")
```

```
1 Nilai Largest of Maximum: 6.99
```

Penjelasan Kode Program

```
1 import numpy as np
2 import skfuzzy as fuzz
3 from skfuzzy.defuzzify import defuzz
4 import matplotlib.pyplot as plt
```

1. Import Library:

- **numpy**: Digunakan untuk komputasi numerik, seperti mendefinisikan semesta pembicaraan (x) dan melakukan operasi matematika.
- **skfuzzy**: Library untuk operasi logika fuzzy. Menyediakan alat untuk membuat fungsi keanggotaan dan melakukan defuzifikasi.
- **matplotlib.pyplot**: Digunakan untuk memvisualisasikan fungsi keanggotaan dan hasil defuzifikasi.

```
1 # Langkah 3: Lakukan defuzifikasi Largest of Maximum (LoM)
2 lom_value = defuzz(x, mfx, mode='lom')
```

2. Lakukan Defuzifikasi Largest of Maximum (LoM):

- Fungsi `defuzz` dari `skfuzzy.defuzzify` digunakan untuk menghitung nilai LoM.
- `mode= 'lom'` menentukan bahwa metode Largest of Maximum digunakan.
- Metode LoM memilih nilai

$$x$$

terbesar di mana fungsi keanggotaan mencapai nilai maksimumnya (

$$\mu_{textmax}$$

).

- Baris kode ini adalah inti dari proses defuzifikasi LoM dalam program ini. Fungsi `defuzz` mengambil semesta pembicaraan (x), fungsi keanggotaan ($\mu_f x$), dan metode defuzifikasi ('lom') sebagai *input*, dan menghasilkan nilai *crisp* LoM.